MISB MOTION IMAGERY STANDARDS BOARD

Engineering Guideline

Video Moving Target Indicator Local Data Set

MISB EG 0903.0

3 September 2009

1 Scope

This Engineering Guideline (EG) defines a Local Data Set (LDS) that may be used to deliver Video Moving Target Indicator (VMTI) metadata in accordance with SMPTE (Society of Motion Picture Television Engineers) 336M. This EG also lays out the relationship between the VMTI LDS and other relevant Standards, and gives implementation guidance for the VMTI LDS.

The intent is to provide VMTI metadata to downstream clients for the purpose of populating Situational Awareness and Common Operating Pictures, generating VMTI overlays on video players and for input to tracking and data fusion systems (e.g. STANAG 4676 compliant systems). In the interests of data efficiency, the VMTI LDS only includes elements relevant to VMTI systems and that are not available in any other Universal or Local data set. For example it does not include or repeat the sensor model parameters detailed in Standard 0601 UAS Datalink LDS stream.

The VMTI LDS provides the option to include a timestamp for use by VMTI systems that output standalone EG 0903 data. Where the VMTI LDS stream is multiplexed into a Standard 0601, it is expected that the VMTI LDS timestamp be used to synchronize the multiplexing with the Standard 0601 stream but not be included in the Standard 0601 stream. The timestamp within the Standard 0601 stream takes precedence.

This principle is extended to the error detection checksum. The VMTI LDS mandates the use of a checksum if sent standalone, but once multiplexed into a broader Standard 0601 stream the VMTI LDS checksum should be deleted in deference to the Standard 0601 checksum.

Every effort has been made to not repeat metadata elements that are found in other LDS's, particularly Standard 0601. Some metadata elements (VMTI Source Sensor, VMTI HFOV and VFOV) in the VMTI LDS are similar to those in Standard 0601 LDS (Image Source Sensor, HFOV and VFOV). This occurs where the VMTI process is run on different video to that described by and/or included with the Standard 0601 data. For example, when infrared (IR) video is described by the Standard 0601 and multiplexed into the MPEG-2 TS but wide field of view video from a bore-sighted visible electro-optic (EO) camera is input to the VMTI process.

2 References

This Engineering Guideline references the following documents and standards:

2.1 Normative References

DMA TM8358.1: Datums, Ellipsoids, Grids, and Grid Reference Systems, 20 September 1990.

IEEE POSIX Standard IEEE 1003.

ISO/IEC 1318-1:2000.

MISB Standard 0807.1, 9 December 2008

SMPTE 336M-2007: Data Encoding Protocol Using Key-Length-Value.

SMPTE RP 210.11: KLV Metadata Dictionary.

2.2 Informative References

MISB EG 0104.5: Predator UAV Basic Universal Metadata Set.

MISB Standard 0601: UAS Datalink Local Data Set.

MISB Standard 0102.5: Security Metadata Universal and Local Data Sets for Digital Motion Imagery.

MISB RP 0103.1: Timing Reconciliation Universal Metadata Set for Digital Motion Imagery.

MISB Standard 0604: Time Stamping Compressed Motion Imagery.

3 Introduction

MISB Standard 0601 has become the accepted standard Local Data Set within Defence agencies for the transmission of metadata elements within motion imagery streams. Standard 0601 includes numerous individual elements and a few local data subsets but, to date, none of these elements or subsets allow the effective inclusion of VMTI metadata.

The purpose of this Engineering Guideline is to define a VMTI LDS which can be transmitted stand alone or embedded within a Standard 0601 data stream. TAG ID 74 from Standard 0601 has been identified for the VMTI LDS.

VMTI is information derived from video and is meaningful as a product in its own right. Therefore, the VMTI LDS is designed to nest within 0601 packets in a transport stream which may only contain information unique to supporting VMTI, that is, contain no video nor unrelated metadata elements (e.g. de-icing).

4 Implementation Conventions

The VMTI LDS is designed to support a large gamut of systems ranging from those producing thousands of moving targets to those producing great detail about a small number of targets. Its design seeks to use bandwidth efficiently and not replicate or duplicate tags or information. It does not include information that can be calculated, derived or associated using other information already defined.

In an effort to balance generality, efficiency and flexibility, data to support applications that add complexity for all other cases yet occur very rarely have been omitted. The expense of including elements to support statistical outlier cases has been considered in terms of the complexity cost increase for all other cases. Similarly, consideration was given to the best design to support the inclusion of VMTI data generated simultaneously from multiple sensors.

4.1 Embedded LDS

The VMTI LDS contains target unique information within an embedded VTargetSeries which, in itself, contains one or more VTarget[1] Packs. The metadata associated with individual targets is delivered using these VTarget Packs (one for each target). There are two elements in each VTarget Pack which, by definition of a pack, are mandatory. The first element of the VTarget Pack is a BER-OID encoded value to convey the Target ID number of the target. The second element is a group of one or more TLV triplets to convey information about the target. No single TLV triplet is mandatory but at least one must be present.

The TLV triplets may refer to individual value and/or sub-ordinate data structures as nested LDS. These LDS include the VMask, VObject, VFeature and VTracker local data sets; currently listed but not elaborated. The VMTI LDS, VTargetSeries and VTarget Pack are designed to have functional implementation prior to the definition of these sub-ordinate embedded LDS. It is envisaged that the definition, content and implementation of these and other LDS can develop over time and be included in future revisions.

An example VMTI LDS packet is shown in Figure 4-1. The packet begins with the Tag for the VMTI LDS within the UAS LDS (Tag 74) and is followed by a length value which is the sum of the lengths of all elements it contains. In this example the VMTI LDS contains one tag 5 element and a VTargetSeries (Tag 101) containing two VTarget Packs, the first of which has a length, a BER-OID encoded Target ID # and a Pixel Number element (Tag 1, Length 1). The second VTarget Pack is similar except that it also contains target latitude and longitude elements.

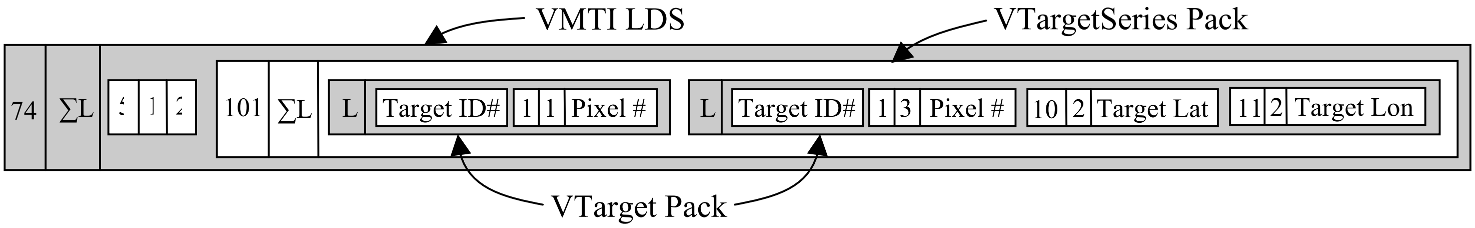

Figure 4-1: Example VMTI Local Data Set Packet

[1] It is assumed that the LDS or Pack name is unique at all levels; therefore references are to VTarget Pack rather than VMTI.VTarget Pack.

4.2 Variable Length Values

The VMTI LDS pioneers the use of variable length values where the length field defines the size of the value payload up to a specified maximum. This technique is mandated for the VMTI LDS and all subordinate structures in order to minimize bandwidth overheads associated with the transmission of the data. Upon ingest to databases, variable length values may be expanded into standard coding structures (for example UINT24 to UINT32) by padding the input bytes with zeros from the MSB, as required.

The VMTI LDS contains some elements that are signed integers. When transitioning a signed integer from standard coding structures to a fewer number of bytes for streaming, care must be taken to preserve both the magnitude and sign of the value. This also applies in reverse.

This version (EG 0903.0) contains no floating point values. If floating point values are added in future revisions, the payload length must be fixed, that is, it can not be variable.

4.3 System Support

All simultaneous VMTI processes from different sensors require individual VMTI LDS. That is, a VMTI LDS should not contain a mixture of moving targets detected by different sensors.

4.3.1 Co-located Bore-sighted Sensors

VMTI hits from co-located, bore-sighted sensor systems may be included as separate VMTI LDS within a single 0601 stream. These are distinguished in the 0601 pack by the VMTI Source Sensor field and horizontal field of view (HFOV) field (some VMTI systems may not know these fields). Thus HFOV and VFOV reside in the VMTI LDS rather than the VTarget Packs

Sophisticated VMTI systems may use the same target ID number to identify a common target detected by different sensors and retain the use of same target id # temporally. Also, downstream processes (e.g. trackers and fusion systems) may re-number the Target ID #'s to correlate a common target.

A system with three or more bore-sighted imagers within a single turret may send a video stream from one camera with a given field of view synchronously with VMTI hits from two different sensors with different fields of view. For example, an L-3 Wescam MX-20 system may have recorded the "video–in-control" stream which may be imaging using an electro-optic narrow (EON) FOV camera and include VMTI hits from both the video electro-optic wide (EOW) FOV and the infrared (IR) cameras. This paradigm is supported by the separate VMTI LDS for each of the IR and EOW cameras within a single 0601 stream multiplexed with the transport stream containing the EON video.

4.3.2 Independent Sensors

Independent systems require separate 0601 streams. The extra elements required in the VMTI LDS to support multiple non-bore-sighted sensors disproportionally increases bandwidth requirements for inclusion within a single 0601 packet. Given that Standard 0601 does not support such cases anyway, the most appropriate solution is to generate individual VMTI LDS and Standard 0601 streams for these sensors.

4.4 Bandwidth

Bandwidth management is an important issue affecting all systems. It is of particular importance for VMTI systems which produce a large amount of dynamic data at frame rates. Thus careful consideration has been given to efficient representation of the data.

In its simplest implementation, the VMTI LDS may send thousands of targets to downstream systems by providing a Target ID# and the target centroid Pixel Number (X and Y coordinates) within the video frame for each target. At the other end of the scale, the VMTI LDS has scope to include multiple features about each target, image chips of the target, tracking information about the target and numerous descriptive elements. The bandwidth overheads required to include all this information are very large – especially at 60 frames per second or higher.

The VMTI LDS includes a large number of elements that may not necessarily be available from VMTI processors. Many of these elements are those normally generated by surveillance management systems. These include all geographic related elements (latitudes, longitudes, velocities etc). The prime function of VMTI systems is to process video frames. It is expected that downstream processes will 'value add' such elements to a basic VMTI LDS stream.

Bandwidth implications and bandwidth management must be considered in the design of systems that generate VMTI LDS data. Reduction of metadata to a minimal configuration is recommended. Although the LDS provides scope for many elements it is undesirable to populate elements just because tags exist to support the data.

5 VMTI LDS Universal Keys

The 16-byte Universal Key for this VMTI Local Data Set is:
06 0E 2B 34 02 0B 01 01 0E 01 03 03 06 00 00 00.

VMTI LDS
Key: 06.0E.2B.34.02.0B.01.01.0E.01.03.03.06.00.00.00
Release Date:
Description:

A key history is provided below as a way to track the keys used in engineering and development.

No key history to date.

6 LDS Packet Structure

Local Data Sets are more bit-efficient than individual keys and also allow for a great deal of flexibility in implementation, allowing users to tailor implementations to their specific and changing needs. This section describes how to create a Local Data Set in accordance with SMPTE 336-M.

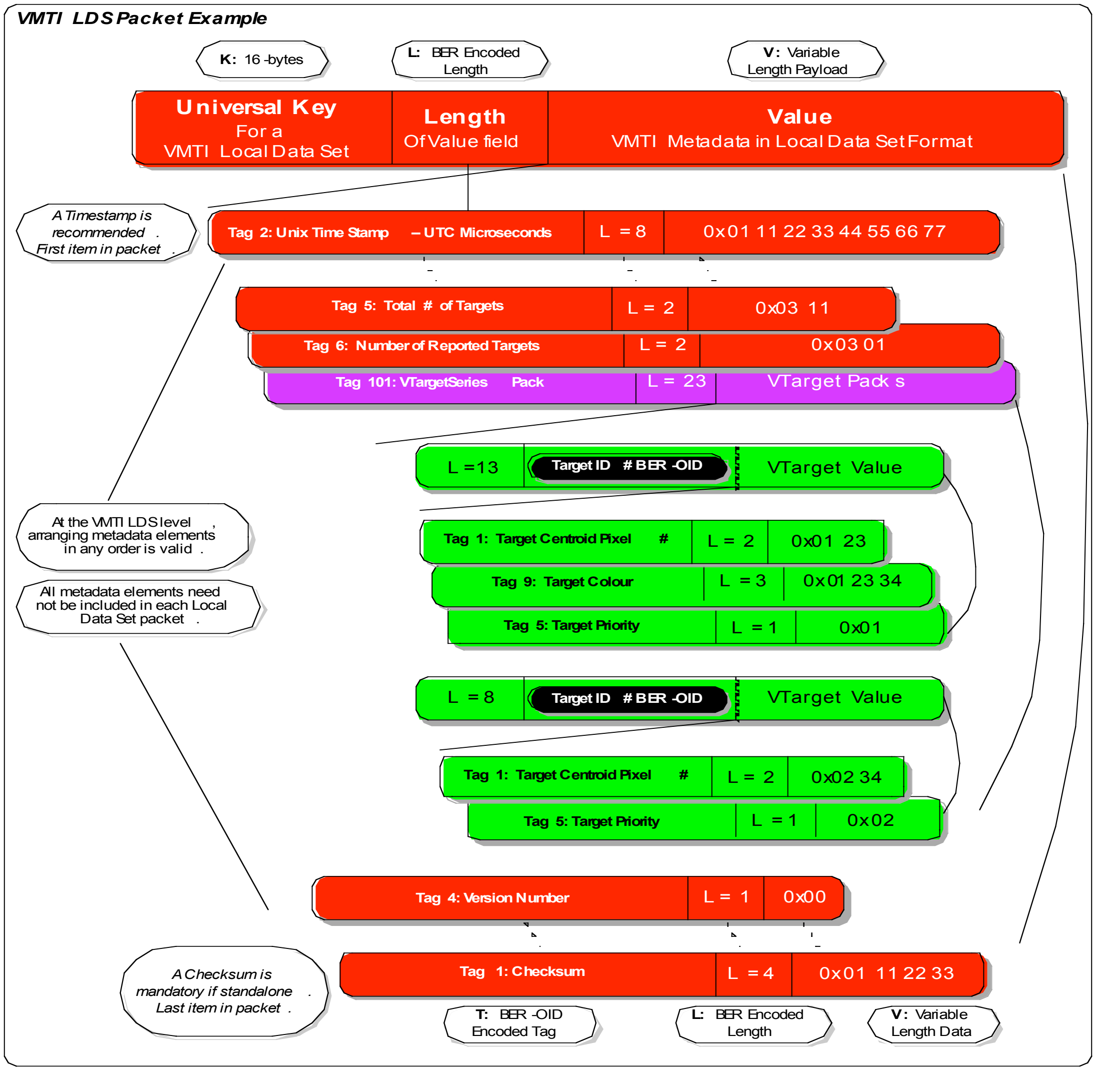
VMTI LDS Packet Example
K: 16 -bytes
L: BER Encoded Length
V: Variable Length Payload
Universal Key
For a
VMTI Local Data Set
Length
Of Value field
Value
VMTI Metadata in Local Data Set Format
A Timestamp is recommended . First item in packet .
Tag 2: Unix Time Stamp – UTC Microseconds
L = 8
0x01 11 22 33 44 55 66 77
Tag 5: Total # of Targets
L = 2
0x03 11
Tag 6: Number of Reported Targets
L = 2
0x03 01
Tag 101: VTargetSeries Pack
L = 23
VTarget Pack s
L =13
Target ID # BER -OID
VTarget Value
At the VMTI LDS level , arranging metadata elements in any order is valid .
All metadata elements need not be included in each Local Data Set packet .
Tag 1: Target Centroid Pixel #
L = 2
0x01 23
Tag 9: Target Colour
L = 3
0x01 23 34
Tag 5: Target Priority
L = 1
0x01
L = 8
Target ID # BER -OID
VTarget Value
Tag 1: Target Centroid Pixel #
L = 2
0x02 34
Tag 5: Target Priority
L = 1
0x02
Tag 4: Version Number
L = 1
0x00
A Checksum is mandatory if standalone . Last item in packet .
Tag 1: Checksum
L = 4
0x01 11 22 33
T: BER -OID Encoded Tag
L: BER Encoded Length
V: Variable Length Data

Figure 6-1: Example VMTI Local Data Set Packet

Figure 6-1 shows the general format of how the VMTI LDS is configured. It is recommended but not mandatory that each VMTI LDS packet contain a Unix-based Coordinated Universal Time (UTC) time stamp that represents the time of birth of the metadata within the LDS packet. If the VMTI LDS pack resides inside a parent 0601 pack then the checksum should be omitted in deference to the 0601 checksum. However, it is mandatory that a checksum metadata item be included in each standalone VMTI LDS pack.

Any combination of metadata items can be included in a VMTI Local Data Set packet. The items within the VMTI LDS can be arranged in any order except that the Unix-based UTC time stamp, if present, should come first and the checksum should come last.

6.1 Bit and Byte Ordering

All metadata is represented using big-endian (Most Significant Byte (MSB) first) encoding. Bytes are big-endian bit encoded (most significant bit (msb) first).

6.2 Tag and Length Field Encoding

According to SMPTE 336M rules, the 6th byte of the 16 byte key[2] for an LDS dictates the encoding to be used for the tag and length fields of the LDS. In the case of the VMTI LDS key, byte 6 is 0x0B. Accordingly the VMTI LDS metadata item tags are encoded using Basic Encoding Rules – Object Identification (BER-OID) and all length fields are encoded using basic encoding rules (BER) for either short or long form encoding of octets. BER length encoding for the length fields provides the greatest level of flexibility for variable length data contained within a KLV packet.

In the case of the current sub-ordinate LDS within the VMTI LDS, namely the VMask, VObject, VFeature and VTracker LDS, byte 6 of their keys is 0x03. Accordingly the metadata item tags for these LDS are a single byte value with no encoding, that is, the maximum number of tags is 255. The length fields of these sub-ordinate LDS are encoded using BER ,that is, identical to the parent VMTI LDS length fields. In the interests of preserving consistency, it is recommended that any sub-ordinate LDS defined in future versions of the VMTI LDS, also set byte 6 of the key to 03.

The second element of the VTarget Pack is a group of one or more TLV triplets. The tags for these triplets shall be single byte values with no length encoding and the length fields shall be encoded using BER.

In practice, the majority of metadata lengths in an LDS packet (and indeed the VTarget Pack) will use the short form of BER encoding which requires only a single byte to represent the length. The length of the entire LDS packet, however, is often represented using the long form of length encoding since the majority of packets have a payload larger than 127 bytes. The key for the entire LDS packet is always 16 bytes. The length of a single packet is represented by 2 bytes whenever the payload portion of the LDS packet is larger than 127, but less than 256 bytes. BER-OID encoding and BER short and long form encoding are discussed in the subsections that follow.

The VMTI LDS includes a special case where the first element of the VTarget Pack – the Target ID # - is a mandated value with no Tag or Length. The value is BER-OID encoded thus defining its length.

See SMPTE 336M Section 4.2 for further details.

6.2.1 BER-OID Encoding Example

BER-OID encoding allows values of any length to be encoded together with information about the length of the value. The value is encoded using a series of binary octets (bytes) where the 1st bit (msb) of each byte indicates whether or not that byte is the last byte in the series. If the 1st bit

[2] See SMPTE 336M-2007 Section 6.3

is set (1) then the remaining 7 bits form part of the value but there are more bytes to follow. If the 1st bit is not set (0) then the next 7 bits are the least significant 7 bits of the value and this is the last byte in the series. As indicated by byte 6 in the VMTI LDS Key (0B) all VMTI LDS Tags shall be BER-OID encoded. In the vast majority of cases Tag numbers will be less than 127 and only one byte will be required to encode them.

The VMTI LDS contains one or more VTarget Packs. All VTarget Packs are grouped together into a single element named "VTarget Series". A value only, BER-OID encoded Target ID # is the first element of each VTarget Pack. Examples of BER-OID encoding for Tag numbers of 101 and 289 are shown in Figure 6-2 and Figure 6-3 respectively.

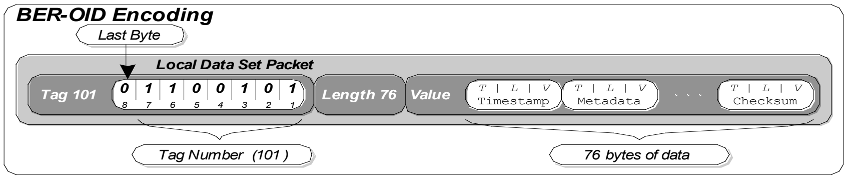

Figure 6-2 BER-OID Encoding of Tag 101

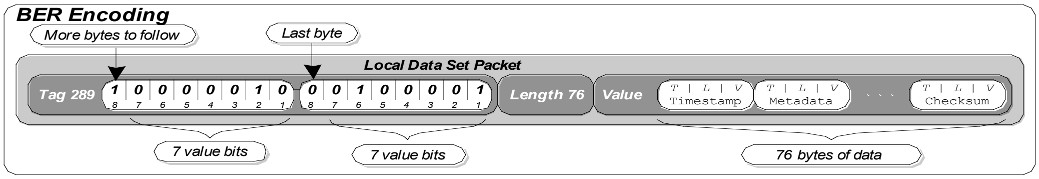

Figure 6-3 BER-OID Encoding of Tag 289

6.2.2 BER Short Form Length Encoding Example

For LDS packets and data elements shorter than 128 bytes, the length field is encoded using the BER short form. Length fields using the short form are represented using a single byte (8 bits). If the most significant bit is set, it signals that the long form is being used. If the most significant bit is not set, it signals that the short form is being used. The last seven bits depict the number of bytes that follow the BER encoded length. An example LDS packet using a short form encoded length is shown in **Error! Reference source not found.**4.

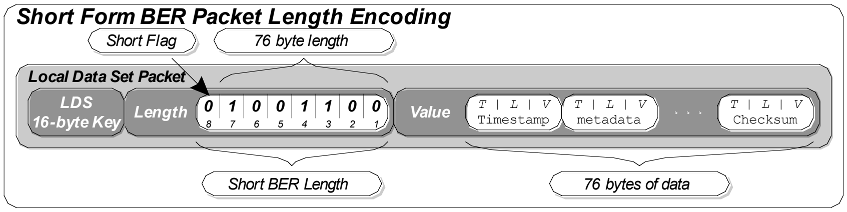

Figure 6-4 Example Short Form Length Encoding

Although this example depicts the length field of the entire LDS packet, short form BER encoding also applies to the lengths within the LDS packet.

6.2.3 BER Long Form Length Encoding

For LDS packets and data elements longer than 127 bytes, the length field is encoded using the BER long form. The long form encodes length fields using multiple bytes. The first byte indicates long form encoding as well as the number of subsequent bytes that represent the length. The bytes that follow the leading byte are the encoding of an unsigned binary integer equal to the number of bytes in the packet. An example LDS packet using a long form encoded length is shown in **Error! Reference source not found.**5.

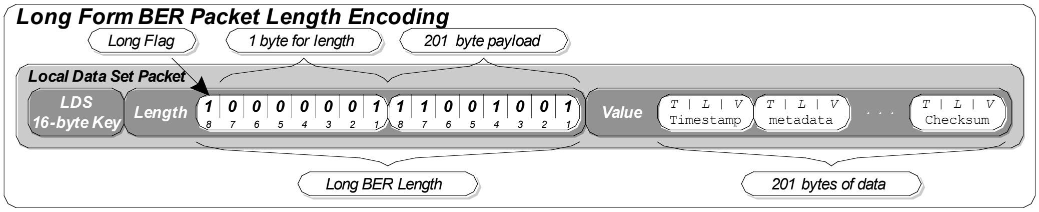

Figure 6-5 Example Long For Length Encoding

Although this example depicts long form BER encoding on the length field of the entire LDS packet, long form BER encoding also applies to the lengths within the LDS packet.

6.3 Time Stamping

It is recommended that every VMTI LDS packet include a Unix-based UTC timestamp as a way to correspond the metadata with a standardized time reference. UTC time is useful to associate metadata with frames, and for reviewing time-critical events at a later date. This section describes how to include a timestamp within a Local Data Set packet.

Metadata sources are coordinated to operate on the same standard time, which is typically GPS derived. The metadata source provides a timestamp for inclusion in an LDS packet and the timestamp assists the accuracy of synchronizing each frame to its corresponding metadata set.

The timestamp tag is User Defined Time Stamp – Microseconds Since 1970. The UTC timestamp (VMTI LDS Tag 02) is an 8 byte unsigned integer that represents the number of microseconds that have elapsed since midnight (00:00:00 UTC), January 1, 1970. This date is known as the UNIX Epoch and is discussed in the IEEE POSIX standard IEEE 1003.1.

6.3.1 Packet Timestamp

It is recommended that the Packet Timestamp be inserted at the beginning of the value portion of a VMTI LDS packet. The timestamp is represented by VMTI LDS Tag 02 (see above) and applies to all metadata in the LDS packet. This timestamp corresponds to the time of birth of all the data within the LDS packet. This time can be used to associate the metadata with a particular video frame and be displayed or monitored appropriately.

An example LDS packet containing a timestamp is show in Figure 6-6.

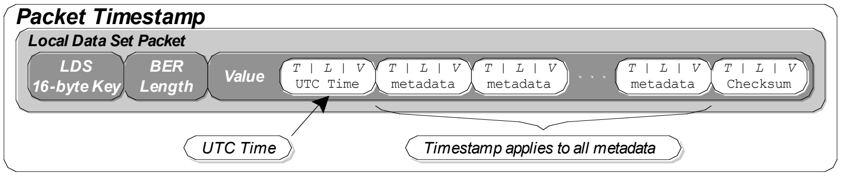

Figure 6-6 Example Packet Timestamp

6.4 Error Detection

To help prevent erroneous metadata from being presented with video, it is mandatory that a 16-bit checksum be included in every standalone VMTI Local Data Set packet as the last item. The checksum is a running 16-bit sum through the entire LDS packet starting with the 16 byte Local Data Set key and ending with summing the length field of the checksum data item. Although mandatory for stand alone VMTI LDS streams, it is recommended that when absorbed into a parent Standard 0601 packet the VMTI LDS checksum be removed in deference to the parent Standard 0601 checksum.

Figure 6-7 shows the data range that the checksum is performed over:

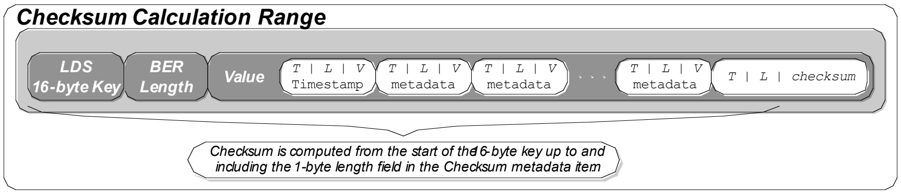
Checksum Calculation Range
LDS 16-byte Key
BER Length
Value
T | L | V Timestamp
T | L | V metadata
T | L | V metadata
. . .
T | L | V metadata
T | L | checksum
Checksum is computed from the start of the 16-byte key up to and including the 1-byte length field in the Checksum metadata item

Figure 6-7 Example Checksum Computation Range

6.5 Tag Formats and Lengths

Tag lengths for the VMTI LDS are BER-OID encoded as indicated by byte 6 in the 16-byte key in accordance with SMPTE 336M.

Tag numbers listed within this document are all in decimal.

6.6 Required Tags

It is mandatory that all stand alone instantiations of the VMTI LDS contain the checksum (VMTI LDS Tag 01). The UNIX-based UTC Timestamp (VMTI LDS Tag 02) is recommended but not mandatory because it is expected that some VMTI systems will not have access to a time source. Data from such systems is still considered useful even if only aligned with the video by time of arrival.

In the interests of bit efficiency, implementations nesting the VMTI LDS within another Local Set should remove the VMTI LDS checksum and UTC timestamp when the external LDS includes a timestamp and checksum.

6.7 Subordinate Data Structures

The VMTI LDS makes use of a subordinate VTargetSeries that, in itself, contains one or more VTarget Packs which, in turn, make use of four subordinate Local Data Sets arranged as follows:

1. VTargetSeries
   a. VTarget Pack
      i. VMask LDS
      ii. VObject LDS
      iii. VFeature LDS
      iv. VTracker LDS

6.7.1 VTarget Pack

VTarget Pack
Key: **06.0E.2B.34.02.05.01.01.0E.01.03.03.07.00.00.00**
Release Date:
Release Version:

The elements of the VTarget Pack are defined in Table 7-3.

The first element of the VTarget Pack must always be the Target ID number. This element is a BER- OID encoded value, that is, it does not have a Tag or Length field associated with it – noting that the BER-OID encoding defines its length.

The second element of the VTarget Pack is a group of one or more TLV triplets to convey information about the target. It is recommended that all instantiations of the VTarget Pack contain, as a minimum, the Target Centroid Pixel Number (Tag 01).

6.7.1.1 VMask LDS

VMask Local Data Set
Key: **06.0E.2B.34.02.03.01.01.0E.01.03.03.08.00.00.00**
Release Date:
Release Version:

The elements of the VMask LDS are undefined; Table 7-4 serves as a place holder.

The VMask LDS is intended to allow the inclusion of metadata elements that define the outline of the detected target. This information can be used to redraw the outline in downstream clients or ‘chip’ the target from the video.

6.7.1.2 VObject LDS

VObject Local Data Set
Key: **06.0E.2B.34.02.03.01.01.0E.01.03.03.09.00.00.00**
Release Date:
Release Version:

The elements of the VObject LDS are undefined; Table 7-5 serves as a place holder.

The VObject LDS is intended to allow the inclusion of an image 'chip' of the target. It is expected that this LDS most commonly be used in bandwidth challenged environments where the operator does not have access to the underlying video stream.

6.7.1.3 VFeature LDS

VFeature Local Data Set
Key: **06.0E.2B.34.02.03.01.01.0E.01.03.03.0A.00.00.00**
Release Date:
Release Version:

The elements of the VFeature LDS are undefined; Table 7-6 serves as a place holder.

The VFeature LDS is intended to allow the inclusion of data that describes the target or features of the target (dents, wheels, number of passengers etc). Descriptive information can range from simple text for a label to complex data sets containing spectral or radiometric data.

6.7.1.4 VTracker LDS

VTracker Local Data Set
Key: **06.0E.2B.34.02.03.01.01.0E.01.03.03.0B.00.00.00**
Release Date:
Release Version:

The elements of the VTracker LDS are undefined; Table 7-7 serves as a place holder.

The VTarget LDS is intended to contain any or all information pertaining to tracking the target. Such information will allow downstream systems (e.g. STANAG 4676) to associate the VMTI targets with tracks from other systems. It is also expected that this LDS will be populated, at least in part, through collaboration with other tracking systems – especially GMTI.

6.8 Key to Tag Mappings

It is required that tags within a Local Data Set map back to valid SMPTE RP 210 (or MISB Standard 0807) keys. When a similar but not identical entry already exists in SMPTE RP210 (or MISB Standard 0807), a new key should be assigned, preferably as a sub-leaf of the existing entry (i.e. replacing the first zero byte of the existing key with 01 or the next available number). When it is not possible to assign a new key, the differences shall be noted in SMPTE RP2009 Groups Register (or the equivalent MISB Standard 0807) and the entry in RP210 shall be informatively noted as Context-Dependent.

It is also possible to map multiple Tags back to a single Key because each instantiation within the LDS carries a unique Tag and is therefore an unambiguous reference.

7 VMTI Local Data Set Table

Table 7-1: VMTI LDS

VMTI LDS						
	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
	06.0E.2B.34.02.0B.01.01 0E.01.03.03.06.00.00.00	VMTI KLV Dictionary	None	N/A	Variable	This is the Universal Key for the VMTI LDS
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes[3]	Notes
01	06.0E.2B.34.04.01.01.01 0E.01.02.03.01.00.00.00	Checksum	None	Uint16	F2	Checksum used to detect errors within a VMTI LDS packet. Defers to Standard 0601 checksum when VMTI LDS packet included inside Standard 0601 packet. Lower 16-bits of summation. Performed on entire VMTI LDS packet, including 16-byte UDS key and 1-byte checksum length.
02	06.0E.2B.34.01.01.01.03 07.02.01.01.01.05.00.00	UNIX Time Stamp	Micro-seconds	Uint64	V8	Microseconds elapsed since midnight (00:00:00 UTC), January 1, 1970 (the UNIX Epoch). Derived from the POSIX IEEE1003.1 standard. Resolution: 1 microsecond.
03	06.0E.2B.34.01.01.01.01 0E.01.02.02.7C.00.00.00	VMTI System Name / Description	String	ISO7	V127	Text string to allow the inclusion of the name and/or description of the VMTI system.
04	06.0E.2B.34.01.01.01.01 0E.01.02.05.04.00.00.00	VMTI LDS Version Number	None	Uint16	V2	Version number of the VMTI LDS document used to generate the VMTI metadata. 0 is pre-release, initial release (090x.0), or test data. 1..65535 corresponds to document revisions 1 through 65535.
05	06.0E.2B.34.01.01.01.01 0E.01.02.03.36.00.00.00	Total Number of Targets Detected in the Frame	None	Uint16	V2	The total number of targets detected in the frame. Range 1 to 65535. 0 represents no targets detected – also implied by no value at all and VMTI LDS is superfluous and should be discarded. 65535 represents 65535 or more detections

[3] Fx = Fixed Length of x Bytes, V = Variable Length with no Maximum, Vx = Variable Length with Maximum of x Bytes

VMTI LDS

	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
06	06.0E.2B.34.01.01.01.01 0E.01.02.03.37.00.00.00	Number of Reported Targets	None	Uint16	V2	The number of targets reported following a culling process. For use, for example, where bandwidth limits the number of targets that can be sent. Range 0 to 65535.
07	06.0E.2B.34.01.01.01.01 0E.01.01.03.1F.00.00.00	Video Frame Number	Frames	Uint24	V3	The video frame number corresponding to the frame in which the targets were detected. Use of the time stamp is preferred but frame number can be used where a time stamp is not available. Range 0 to 2^{24} -1 which equates to approximately 155 hours at 60fps.
08	06.0E.2B.34.01.01.01.01 0E.01.01.02.07.00.00.00	Frame Width	Pixels	Uint24	V3	Width of the video frame in pixels. Range 1 to 16777215 Value of zero is meaningless and should not be used
09	06.0E.2B.34.01.01.01.01 0E.01.01.02.08.00.00.00	Frame Height	Pixels	Unit24	V3	Height of the video frame in pixels. Range 1 to 16777215 Value of zero is meaningless and should not be used
10	06.0E.2B.34.01.01.01.01 04.20.01.02.01.01.00.00	VMTI Source Sensor	String	ISO7	V127	String of VMTI source sensor. E.g.: 'EO Nose', 'EO Zoom (DLTV)', 'EO Spotter', 'IR Mitsubishi PtSi Model 500', 'IR InSb Amber Model TBT', 'LYNX SAR Imagery', 'TESAR Imagery', etc. Value field is Free Text. Maximum 127 characters. Similar to Standard 0601 Tag 11 Used to identify the VMTI process if more than one is run simultaneously.
11	06.0E.2B.34.01.01.01.02 04.20.02.01.01.08.00.00	VMTI Sensor Horizontal Field of View	Degrees	Uint16	V2	Horizontal field of view of imaging sensor input to VMTI process. Required if VMTI process is run on a different imaging sensor to that described by the parent Standard 0601 packet. Can be used with HFOV (Tag 16) from Standard 0601 to scale VMTI X,Y coordinates. Map 0..(2^16-1) to 0..180. Resolution: ~2.7 milli degrees.

VMTI LDS

	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
12	06.0E.2B.34. 01.01.01.07 04.20.02.01.01.0A.01.00	VMTI Sensor Vertical Field of View	Degrees	Uint16	V2	Vertical field of view of imaging sensor input to VMTI process. May be required if VMTI process is run on a different imaging sensor to that included in the MPEG-2 TS. Can be used with VFOV (Tag 17) from Standard 0601 to scale VMTI X,Y coordinates. Typically only required to cater for aspect ratio variation. Map 0..(2^16-1) to 0..180. Resolution: ~2.7 milli degrees.
101	N/A	VTargetSeries	None	N/A	V	Series of target metadata, each of which is a. VTarget Pack as defined in Table 7-2 below. The length field for VTargetSeries is the sum of the lengths of all the contained target metadata. The length field for each VTarget Pack is the size of all elements within that pack including the Target ID #.

Table 7-2 VTarget Pack

VTarget Pack

	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
		VTarget Pack	None	N/A	Variable	This is the Universal Key for the VTarget Pack. This is a Truncation Pack
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
N/A	**N/A**	Target ID Number		Uint16	V3	This element is mandatory and it must come first in the VTarget Pack. It is BER-OID encoded to convey the length but has no Tag or Length field. Range 1..65535
N/A	**N/A**	Target Local Data		Series	Var	This element is optional. Its value is a series of target local data items as defined in the following table

Table 7-3: Target Local Data

Target Local Data						
	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
		Target Local Data	None	N/A	Variable	
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
01	06.0E.2B.34.01.01.01.01 0E.01.02.03.38.00.00.00	Target Centroid Pixel Number	Pixels	Uint40	V5	Defines the position of the target within the video frame in pixels. Range 1 to 2^{40}-1. Numbering commences from one denoting the Top Left pixel Maximum size of frame is 1 Terapixel
02	06.0E.2B.34.01.01.01.01 0E.01.02.03.39.00.00.00	Bounding Box Top Left Pixel Number	Pixels	Uint40	V5	Defines the position of the top left corner of the target bounding box within the video frame in pixels. Range 1 to 2^{40}-1. Numbering commences from one denoting the Top Left pixel Maximum size of frame is 1 Terapixel
03	06.0E.2B.34.01.01.01.01 0E.01.02.03.3A.00.00.00	Bounding Box Bottom Right Pixel Number	Pixels	Uint40	V5	Defines the position of the bottom right corner of the target bounding box within the video frame in pixels. Range 1 to 2^{40}-1. Numbering commences from one denoting the Top Left pixel Maximum size of frame is 1 Terapixel
04	06.0E.2B.34.01.01.01.01 0E.01.02.03.3B.00.00.00	Target Priority	None	Uint8	F1	Priority or validity of target based on criteria within the VMTI system. The target(s) with the highest priority may not have the highest confidence value. Potential for use in limited bandwidth scenarios to only send highest priority targets. Range 1 to 255 where 1 is the highest priority. Multiple targets may have the same priority.
05	06.0E.2B.34.01.01.01.01 0E.01.02.03.3C.00.00.00	Target Confidence Value	None	Uint8	F1	Confidence value of target based on criteria within the VMTI system. The target(s) with the highest confidence may not have the highest priority value. Potential for use in limited bandwidth scenarios to only send highest confidence targets. Range 1 to 255 where 1 is the highest confidence. Multiple targets may have the same confidence value.

06	06.0E.2B.34.01.01.01.01 0E.01.02.03.3D.00.00.00	New Detection Flag / Target History	Frames	Uint16	V2	The number of previous times the same target has been detected. Range 0 to 65535 frames. Where a value of 0 denotes the target as a new detection. Detections are not required to be in consecutive frames.
07	06.0E.2B.34.01.01.01.01 0E.01.02.03.3E.00.00.00	Percentage of Target Pixels	None	Uint8	F1	The percentage of pixels within the bounding box that are detected to be target pixels rather than background pixels. Range 1 to 100 where 100 signifies that the target completely fills the bounding box.
08	06.0E.2B.34.01.01.01.01 0E.01.02.03.3F.00.00.00	Target Colour	None	Uint24	F3	Dominant colour of the target. For use when metadata is transmitted in the absence of the underlying video. RGB colour value. VFeature LDS can be used for more comprehensive colour information.
09	06.0E.2B.34.01.01.01.01 0E.01.02.03.40.00.00.00	Target Intensity	None	Uint24	V3	Dominant Intensity of the target. For use when metadata is transmitted in the absence of the underlying video. Range 0..2^{24}-1. VFeature LDS can be used for more comprehensive temperature information.
10	06.0E.2B.34.01.01.01.01 0E.01.02.03.41.00.00.00	Target Location Latitude Offset	Degrees	Int24	V3	Latitude offset for target from frame centre latitude (Standard 0601). Based on WGS84 ellipsoid. Use with Frame Center Latitude. Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees. Use -(2^{23}) as an "error" indicator. Resolution: ~1.2micro deg, ~0.25 meters at equator. Range: +/- 2136 km.
11	06.0E.2B.34.01.01.01.01 0E.01.02.03.42.00.00.00	Target Location Longitude Offset	Degrees	Int24	V3	Longitude offset for target from frame centre longitude (Standard 0601). Based on WGS84 ellipsoid. Use with Frame Center Longitude. Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees. Use -(2^{23}) as an "error" indicator. Resolution: ~1.2micro deg, ~0.25 meters at equator. Range: +/- 2136 km.
12	06.0E.2B.34.01.01.01.01 0E.01.02.03.43.00.00.00	Target Elevation	Meters	Uint16	V2	Terrain elevation at target location. Map 0..(2^{16}-1) to -900..19000 meters. Resolution: ~0.3 meters.
13	06.0E.2B.34.01.01.01.01 0E.01.02.03.44.00.00.00	Bounding Box Top Left Latitude Offset	Degrees	Int24	V3	Latitude offset for top left corner of target bounding box. Use with Standard 0601 Frame Centre Latitude. Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees. Use -(2^{23}) as an "error" indicator. Resolution: ~1.2micro deg, ~0.25 meters at equator. Range: +/- 2136 km.

14	06.0E.2B.34.01.01.01.01 0E.01.02.03.45.00.00.00	Bounding Box Top Left Longitude Offset	Degrees	Int24	V3	Longitude offset for top left corner of target bounding box. Use with Standard 0601 Frame Centre Longitude. Map $-(2^{23}-1)..(2^{23}-1)$ to +/- 19.2 degrees. Use $-(2^{23})$ as an "error" indicator. Resolution: ~1.2micro deg, ~0.25 meters at equator. Range: +/- 2136 km.
15	06.0E.2B.34.01.01.01.01 0E.01.02.03.46.00.00.00	Bounding Box Bottom Right Latitude Offset	Degrees	Int24	V3	Latitude offset for bottom right corner of target bounding box. Use with Standard 0601 Frame Centre Latitude. Map $-(2^{23}-1)..(2^{23}-1)$ to +/- 19.2 degrees. Use $-(2^{23})$ as an "error" indicator. Resolution: ~1.2micro deg, ~0.25 meters at equator. Range: +/- 2136 km.
16	06.0E.2B.34.01.01.01.01 0E.01.02.03.47.00.00.00	Bounding Box Bottom Right Longitude Offset	Degrees	Int24	V3	Longitude offset for bottom right corner of target bounding box. Use with Standard 0601 Frame Centre Longitude. Map $-(2^{23}-1)..(2^{23}-1)$ to +/- 19.2 degrees. Use $-(2^{23})$ as an "error" indicator. Resolution: ~1.2micro deg, ~0.25 meters at equator. Range: +/- 2136 km.
101	06.0E.2B.34.02.03.01.01 0E.01.03.03.08.00.00.00	VMask LDS (To Be Defined)	None	N/A	V	Local set tag to include a mask for delineating the perimeter of the target. It may be used to extract the target and populate the VObject LDS. Use the VMask Local Set Tags. The length field is the size of all VMask items to be packaged within this tag.
102	06.0E.2B.34.02.03.01.01 0E.01.03.03.09.00.00.00	VObject LDS (To Be Defined)	None	N/A	V	Local set tag to include underlying pixel values for the target. The VObject LDS contains a target chip extracted from the video. Use the VObject Local Set Tags. The length field is the size of all VObject items to be packaged within this tag.
103	06.0E.2B.34.02.03.01.01 0E.01.03.03.0A.00.00.00	VFeature LDS (To Be Defined)	None	N/A	V	Local set tag to include features about the target. More than one VFeature LDS can be included within the VTarget Pack. Use the VFeature Local Set Tags. The length field is the size of all VFeature items to be packaged within this tag.
104	06.0E.2B.34.02.03.01.01 0E.01.03.03.0B.00.00.00	VTracker LDS (To Be Defined)	None	N/A	V	Local set tag to include track information about the target. Use the VTracker Local Set Tags. The length field is the size of all VTracker items to be packaged within this tag.

Table 7-4: VMask LDS

VMask Local Data Set						
	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
	06.0E.2B.34.02.03.01.01 0E.01.03.03.08.00.00.00	VMask LDS	None	N/A	Variable	This is the Universal Key for the VMask LDS
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes

Table 7-5: VObject LDS

VObject Local Data Set						
	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
	06.0E.2B.34.02.03.01.01 0E.01.03.03.09.00.00.00	VObject LDS	None	N/A	Variable	This is the Universal Key for the VObject LDS
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes

Table 7-6: VFeature LDS

VFeature Local Data Set						
	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
	06.0E.2B.34.02.03.01.01 0E.01.03.03.0A.00.00.00	VFeature LDS	None	N/A	Variable	This is the Universal Key for the VFeature LDS
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes

Table 7-7: VTracker LDS

VTracker Local Data Set						
	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes
	06.0E.2B.34.02.03.01.01 0E.01.03.03.0B.00.00.00	VTracker LDS	None	N/A	Variable	This is the Universal Key for the VTracker LDS
Tag ID	Key Value (hex)	Key Name	Units	Format	Length in Bytes	Notes

8 Glossary of Acronyms and Symbols

BER	Basic Encoding Rules
EON	Electro-Optic Narrow
EOW	Electro-Optic Wide
FOV	Field of View
KLV	Key-Length-Value
ISS	Image Source Sensor
LDS	Local Data Set
MISB	Motion Imagery Standards Board
MSB	Most Significant Byte
msb	Most Significant Bit
SMPTE	Society of Motion Picture and Television Engineers
TLV	Tag, Length, Value
UINT	Unsigned Integer
VMTI	Video Moving Target Indicator
VSS	VMTI Source Sensor

9 Individual Tag Descriptions and Conversions and Mappings Between Metadata Types

Metadata items that are common amongst UAS LDS, Predator UDS, and ESD data formats each convey identical information. However, since each metadata format represents the same metadata items differently (e.g. mapped integer, float, string, etc.), the data resolution between format types is different. This section provides descriptions of each VMTI LDS metadata element and conversions and mappings, where applicable between the VMTI LDS and the UDS. None of the VMTI LDS metadata elements have equivalent ESD elements.

Video sensors with frame rates of up to 60 frames per second are used operationally and faster systems are expected in the future. It follows that VMTI LDS packets may be sent at equivalent rates. A distinction has been made between dynamic data and static data in order to reduce the bandwidth overhead required for the transmission of metadata packets at frame rates.

Fields marked with an “x” are to be considered not applicable.

9.1 Tag 1: Checksum

LDS Tag	1	Units	Range	Format
LDS Name	Checksum	None	$0..(2^{16}$-1)	Uint16
UDS Mapped	06 0E 2B 34 04 01 01 01			
Key	0E 01 02 03 01 00 00 00			

Notes	Conversion Formula
- Checksum used to detect errors within a UAV Local Data Set packet. - Lower 16-bits of summation. - Performed on entire LDS packet, including 16-byte UDS key and 1-byte checksum length.	x

Example Value	Example LDS Packet
0x00 00	`[K] [L] [V] = [0d1] [0d2] [0x00 00]`

UDS Key	x
UDS Name	x

Units	Range	Format
x	x	x

Notes
x

UDS Conversion
x

It is mandatory that the checksum be included with each and every standalone VMTI LDS packet and that it should come last in the packet. However, it is recommended that when absorbed into a parent Standard 0601 packet the VMTI LDS checksum be removed in deference to the parent Standard 0601 checksum.

A checksum has not been included for each of the subordinate sets within the VMTI LDS. Bandwidth efficiency is the prime reason for this. It is allowable and conceivable that hundreds of targets will be detected per frame. Frame rates of 60 fps and higher will be commonplace. If included in the subordinate sets (VTarget Pack) a two byte checksum will add approximately 100kbps to the bandwidth overhead for every 100 targets in the frame. However, if no checksum is included with each VTarget Pack hundreds of targets will be declared invalid if even one of them has a transmission error because the only other checksum is the VMTI LDS checksum. Nevertheless those same targets will be transmitted with the next frame one 60th of a second later. If the data link is so poor as to have errors in every frame then this is an issue related to the data link and should be dealt with in context rather than condemn all uses to unnecessary overheads.

– Consider optional checksum – perhaps based on target priority or confidence.

Refer to Standard 0601 for example 16-bit checksum code.

9.2 Tag 2: UNIX Time Stamp

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	2 UNIX Time Stamp Use EG0104 UDS Key	Microseconds	$0..(2^{64}-1)$	Unsigned Integer of Variable Length

Notes	Conversion Formula
- Microseconds elapsed since midnight (00:00:00), January 1, 1970. - Derived from the POSIX IEEE 1003.1 standard. - Resolution: 1 microsecond.	x

Example Value	Example LDS Packet
April 19 2001, 04:25:21 GMT	`[K] [L] [V] = [0d2] [0d8] [0x00 03 82 44 30 F6 CE 40]`

UDS Key	06 0E 2B 34 01 01 01 03 07 02 01 01 01 05 00 00
UDS Name	User Defined Time Stamp - microseconds since 1970

Units	Range	Format
Microseconds	$0..(2^{64}-1)$	Uint64

Notes
- Time Stamp application defined by user. - 64 bit integer which represents the number of microseconds since Jan 1, 1970 UTC. - Derived from the POSIX IEEE 1003.1 standard.

UDS Conversion
x

Unix time, or POSIX time, is a system used to discretely label a scale of time. This system is widely used within systems of differing underlying architectures. Unix time is an encoding of Coordinated Universal Time (UTC) and therefore accounts for the addition (or subtraction) of leap seconds. Leap seconds are used to synchronize the UTC clock metric with the yearly rotation period of the earth about the sun.

It is recommended that the time stamp be included with each and every VMTI LDS packet and that it should come first in the packet. It is also recommended that when absorbed into a parent Standard 0601 packet the VMTI LDS timestamp be removed in deference to the parent Standard 0601 timestamp.

Some VMTI systems may not have access to a GPS time signal. In such circumstances designers are recommended to use the VMTI CPU clock or embedded video time code if available. If no time information is available at all, VMTI LDS packets can be sent without a timestamp. These packets can derive timestamps related to time of arrival at downstream processors / systems and use Tag 7: Video Frame Number (if present) as a frame sequence check.

9.3 Tag 3: VMTI System Name or Description

		Units	Range	Format
LDS Tag	3	String	1..32	ISO7
LDS Name	VMTI System Name/Description			
UDS Mapped Key				

Notes	Conversion Formula
- Name or description of the VMTI system. - Value field is free text. - Maximum 32 characters.	x

Example Value	Example LDS Packet
DSTO_ADSS_VMTI	`[K] [L] [V] = [0d3] [0d14] [0x44 53 54 4F 5F 41 44 53 53 5F 56 4D 54 49]`

UDS Key	
UDS Name	

Units	Range	Format
String	1..32	ISO7

Notes

UDS Conversion
x

The VMTI system name or description field is used to identify the system that produced the VMTI targets. The system name/description element is static and as such, an update rate in the order of once per 2 seconds is considered adequate.

9.4 Tag 4: VMTI LDS Version Number

LDS Tag LDS Name UDS Mapped Key	4 VMTI LDS Version Number	Units	Range	Format
		None	0..65535	Unsigned Integer of Variable Length
Notes		Conversion Formula		
- Version number of the VMTI LDS document used to generate the VMTI metadata. - 0 is pre-release, initial release (090x.0), or test data. - 1...65535 corresponds to document revisions 1 through 65535.		x		

Example Value	Example LDS Packet
2	`[K] [L] [V] = [0d4] [0d1] [0x02]`

UDS Key UDS Name	

Units	Range	Format
None	0..65535	Uint16

Notes

UDS Conversion
x

The VMTI LDS version number serves to notify to downstream clients the version of the VMTI LDS used to encode the VMTI metadata. The VMTI LDS version number element is static and, as such, an update rate in the order of once per 2 seconds is considered adequate.

9.5 Tag 5: Total Number of Targets Detected in the Frame

LDS Tag	5	Units	Range	Format
LDS Name	Total # of Targets Detected in the Frame	None	1..65535	Unsigned Integer of Variable Length
UDS Mapped Key				

Notes	Conversion Formula
- Total number of moving targets detected in the frame. - Range 1...65535. - 0 represents no targets detected - Also implied by no value at all in which case VMTI LDS is superfluous and should be discarded.	x

Example Value	Example LDS Packet
14	`[K] [L] [V] = [0d5] [0d1] [0x0E]`

UDS Key UDS Name	

Units	Range	Format
None	1..65535	Uint16

Notes

UDS Conversion

The total number of moving targets detected in the video frame provides clients with pertinent information. This value represents the total number of moving objects detected by the VMTI process in the current frame. This is particularly relevant when the number of targets reported (Tag 6) is less than the total number detected in the frame. This circumstance may arise when there are a large number of detected targets and a culling process has reduced the number of targets included in the VMTI LDS packet. Alternatively, the motion imagery provided in the MPEG-2 TS may be derived from a different (bore-sighted) sensor to that used to detect moving targets. For example, the MPEG-2 TS contains imagery from a spotter sensor but the VMTI process was run on a wide angle sensor.

This metadata element is dynamic and, if present, should be sent with each and every VMTI LDS packet.

9.6 Tag 6: Number of Reported Targets

LDS Tag	6
LDS Name	Number of Reported Targets
UDS Mapped Key	

Units	Range	Format
None	1..65535	Unsigned Integer of Variable Length

Notes	Conversion Formula
- Number of moving targets reported from the frame. - Number of Reported targets = Total Number of Targets – Number of Culled Targets - Culling process usually linked to priority value or confidence value. - Range 1..65535.	x

Example Value	Example LDS Packet
28	[K] [L] [V] = [0d6] [0d1] [0x1C]

UDS Key	
UDS Name	

Units	Range	Format
None	1..65535	Uint16

Notes

UDS Conversion

The number of reported targets in the frame is required when one or more detected targets are not included in the VMTI LDS. It may be necessary (for bandwidth efficiency) to only report a subset of the detected targets. A culling process may be performed at the destination (for example, by display software) or at the source by the VMTI system itself. Culling may also take place when stand alone VMTI LDS packets are absorbed into Standard 0601 packets. In all cases it is expected that the culling will be undertaken according to priority values (VTarget Pack Tag 4) and/or confidence values (VTarget Pack Tag 5).

This metadata element is dynamic and, if present, should be sent with each and every VMTI LDS packet.

9.7 Tag 7: Video Frame Number

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	7 Video Frame Number	Frames	0..(2^{24}-1)	Unsigned Integer of Variable Length

Notes	Conversion Formula
- The video frame number corresponding to the frame in which the targets were detected. - Use of the time stamp (Tag 2) is preferred but frame number can be used where a time stamp is not available. - Range 0 to 224 -1 which equates to approximately 155 hours at 60fps.	x

Example Value	Example LDS Packet
78000	`[K] [L] [V] = [0d7] [0d3] [0x01 30 B0]`

UDS Key UDS Name	

Units	Range	Format
Frames	0..(2^{24}-1)	Uint24

Notes

UDS Conversion
x

The video frame number is only required when a time stamp is not available and, even then, only useful if the video source includes a frame number. It is expected that most VMTI systems will be provided video directly from the sensor rather than from a video encoder. This is for two reasons; firstly, the VMTI process is likely to produce better results on uncompressed video and secondly, latencies can be minimized by running the VMTI and encoding processes in parallel rather than in sequence.

In the absence of a frame number in the raw video source, frame capture devices typically number frames from commencement of the capture and begin with frame 1. If the processes are parallel, the VMTI process will require its own capture process – independent to the encoder capture process and the frame numbers of each will not be synchronous. Nevertheless, there will be cases where the VMTI process is run on encoded video. Under these circumstances and in the absence of a timestamp, the VMTI system should use the source frame number to populate the video frame number element.

This metadata element is dynamic and, if present, should be sent with each and every VMTI LDS packet. It is expected that this will be used as a temporary metadata element to ensure the integrity of the frame order until discarded in deference to the introduction of Tag 2: UNIX Time Stamp derived from the time of arrival at downstream processors/systems.

9.8 Tag 8: Frame Width

LDS Tag	8	Units	Range	Format
LDS Name	Frame Width	Pixels	1..16777215	Unsigned Integer of Variable Length
UDS Mapped Key				

Notes	Conversion Formula
- Width of the video frame in pixels. - Range 1 to 16777215 - Value of zero is meaningless and should not be used - Maximum 3 bytes	x

Example Value	Example LDS Packet
1920	[K] [L] [V] = [0d8][0d2][0x07 80]

UDS Key	
UDS Name	

Units	Range	Format
Pixels	1..16777215	Uint24

Notes

UDS Conversion

Frame width has been included for two reasons. First, efficiency can be realized by the use of pixel number rather than X, Y coordinates to locate the target within the video frame. The pixel number calculation relies upon frame width. Second, two cameras of different frame size may reside within the one turret (e.g. High Definition spotter scope with Standard Definition wide angle). The underlying video in the transport stream includes frame size information but when the underlying video is different from the video on which the VMTI process was run (or a metadata only stream is transmitted), then this Tag can be used to include the frame width.

This metadata element is static and as such, an update rate in the order of once per 2 seconds is considered adequate.

9.9 Tag 9: Frame Height

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	9 Frame Height	Pixels	1..16777215	Unsigned Integer of Variable Length

Notes	Conversion Formula
- Height of the video frame in pixels. - Range 1 to 16777215 - Value of zero is meaningless and should not be used - Maximum 3 bytes	x

Example Value	Example LDS Packet
1080	[K] [L] [V] = [0d9][0d2][0x04 38]

UDS Key UDS Name	

Units	Range	Format
Pixels	1..16777215	Uint24

Notes

UDS Conversion

See description of frame width under Tag 8.

This metadata element is static and as such, an update rate in the order of once per 2 seconds is considered adequate.

9.10 Tag 10 VMTI Source Sensor

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	10 VMTI Source Sensor	String	1..127	ISO7

Notes	Conversion Formula
- String of image source sensor. E.g.: 'EO Nose', 'EO Zoom (DLTV)', 'EO Spotter', 'IR Mitsubishi PtSi Model 500', 'IR InSb Amber Model TBT', 'LYNX SAR Imagery', 'TESAR Imagery', etc. - Value field is Free Text. - Maximum 127 characters. - Similar to Standard 0601 Tag 11 - Could also be used to identify the VMTI process if more than one is run simultaneously.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d10][0d1][0x1B]

UDS Key UDS Name	06 0E 2B 34 01 01 01 01 04 20 01 02 01 01 00 00 Image Source Device

Units	Range	Format
String	1..127	ISO7

Notes

UDS Conversion

This metadata element is static and as such, an update rate in the order of once per 2 seconds is considered adequate Example VMTI Source Sensor

A sample imaging source sensor for the VMTI process is shown in the figure below:

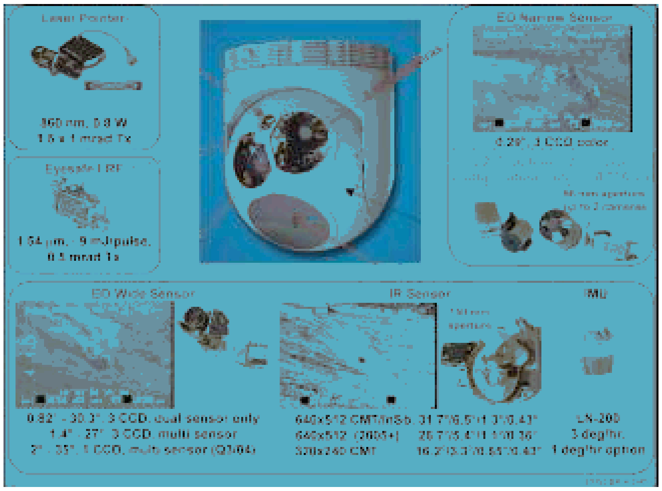

Figure 9-1: Sample Imaging Sensor

9.11 Tag 11 Sensor Horizontal Field of View

LDS Tag	11	Units	Range	Format
LDS Name	VMTI Sensor Horizontal Field of View	Degrees	0..180	Unsigned Integer of Variable Length
UDS Mapped Key				

Notes

- Horizontal field of view of imaging sensor input to VMTI process. Only required if VMTI process is run on a different imaging sensor to that described by the parent Standard 0601 packet.
- Can be used with HFOV (Tag 16) from Standard 0601 to scale VMTI X,Y coordinates.
- Map 0..(2^16-1) to 0..180.
- Resolution: ~2.7 milli degrees.

Conversion Formula

$$\mathrm{VMTI\_LDS\_dec} = \left(\frac{\mathrm{VMTI\_LDS\_range}}{\mathrm{uint\_range}} * \mathrm{VMTI\_LDS\_uint} \right)$$

$$\mathrm{VMTI\_LDS\_11\_dec} = \left(\frac{180}{65535} * \mathrm{VMTI\_LDS\_11} \right)$$

Example Value	Example LDS Packet
144.5713 Degrees	[K] [L] [V] = [0d11][0d2][0xCD 9C]

UDS Key	
UDS Name	

Units	Range	Format
None	0..2^{24}-1	Uint24

Notes

Could be mapped to the UAS FOV-Horizontal element but could cause confusion between this HFOV and the HFOV of the sensor described by the Standard 0601 stream. Once outside the VMTI LDS there would be no way of distinguishing the two values

UDS Conversion

This value should only be populated if the video input to the VMTI process is different to that streamed with the Standard 0601 data. Under these circumstances the ratio (k_x) between the HFOV value in the VMTI LDS and that in the Standard 0601 stream can be used to scale the VMTI X,Y coordinates for display in the streamed video display. The scaling is given by:

$$x_2 = k_x \left(x_1 - \left(\frac{Frame\_Width}{2} \right) \right) + \frac{Frame\_Width}{2}$$

Where: x_1 is the original X coordinate of the target extracted from the target pixel number.
x_2 is the scaled X coordinate of the target in the video.
k_x is the scaling factor calculated according to the following equation:

$$k_x = \frac{\tan\left(\frac{1}{2}\theta_{H1}\right)}{\tan\left(\frac{1}{2}\theta_{H2}\right)}$$

Where: θ_{H1} is the HFOV of the original (VMTI) sensor.

θ_{H2} is the HFOV of the sensor to which the targets are being scaled.

Example to be added

It is expected that in most cases k_x will be sufficient to perform scaling in both the X and Y directions (Y scaling uses frame height rather than frame width – see Tag 12). The VFOV element in VMTI LDS 12 is provided for those cases where the aspect ratio of the two sensors is different (for example 4:3 and 16:9).

These equations are valid if the frame width of the two sensors is the same. If not, the equation becomes more complex.

When the streamed video is from a narrow field of view sensor and the VMTI process is run on video from a bore-sighted wide field of view sensor it can be expected that there are moving targets are outside the frame of the video player. Under these circumstances the video player could add a blank border around the active video and present the moving targets to the operator albeit without the underlying video. Alternatively, the video player could present highlights around the perimeter of the frame to indicate movers outside the frame and their radial direction from the bore-sight.

This metadata element is dynamic and, if present, should be sent with each and every VMTI LDS packet.

9.12 Tag 12 VMTI Sensor Vertical Field of View

		Units	Range	Format
LDS Tag	12	Degrees	0..180	Unsigned Integer of Variable Length
LDS Name	VMTI Sensor Vertical Field of View			
UDS Mapped Key				

Notes

- Vertical field of view of imaging sensor input to VMTI process. Only required if VMTI process is run on a different imaging sensor to that described by the parent Standard 0601 packet.
- Can be used with H\VFOV (Tag 17) from Standard 0601 to scale VMTI X,Y coordinates. Typically only required to cater for aspect ratio variation.
- Map 0..(2^16-1) to 0..180.
- Resolution: ~2.7 milli degrees.

Conversion Formula

$$\text{VMTI\_LDS\_dec} = \left(\frac{\text{VMTI\_LDS\_range}}{\text{uint\_range}} * \text{VMTI\_LDS\_uint} \right)$$

$$\text{VMTI\_LDS\_12\_dec} = \left(\frac{180}{65535} * \text{VMTI\_LDS\_12} \right)$$

Example Value	Example LDS Packet
152.6436 Degrees	[K] [L] [V] = [0d12][0d2][0xD9 17]

UDS Key UDS Name	

Units	Range	Format
x	x	x

Notes

UDS Conversion

This value should only be populated if the video input to the VMTI process is different to that streamed with the Standard 0601 data. Under these circumstances the ratio (k_y) between the VFOV value in the VMTI LDS and that in the Standard 0601 stream can be used to scale the VMTI Y coordinate for display in the streamed video display. The scaling is given by:

$$y_2 = k_y \left(y_1 - \left(\frac{Frame\_Height}{2} \right) \right) + \frac{Frame\_Height}{2}$$

Where: y_1 is the original Y coordinate of the target extracted from the target pixel number.
y_2 is the scaled Y coordinate of the target in the video.
k_y is the scaling factor calculated according to the following equation:

$$k_y = \frac{\tan\left(\frac{1}{2}\theta_{V1}\right)}{\tan\left(\frac{1}{2}\theta_{V2}\right)}$$

Where: θ_{V1} is the VFOV of the original (VMTI) sensor.

θ_{V2} is the VFOV of the sensor to which the targets are being scaled.

Example to be added

It is expected that in most cases k_x (see Tag 11) will be sufficient to perform scaling in both the X and Y directions. The VFOV element in VMTI LDS 12 is provided for those cases where the aspect ratio of the two sensors is different (for example 4:3 and 16:9).

These equations are valid if the frame height of the two sensors is the same. If not, the equation becomes more complex.

When the streamed video is from a narrow field of view sensor and the VMTI process is run on video from a bore-sighted wide field of view sensor it can be expected that there are moving targets are outside the frame of the video player. Under these circumstances the video player could add a blank border around the active video and present the moving targets to the operator albeit without the underlying video. Alternatively, the video player could present highlights around the perimeter of the frame to indicate movers outside the frame and their radial direction from the bore-sight.

This metadata element is dynamic and, if present, should be sent with each and every VMTI LDS packet.

9.13 Tag 101: VTargetSeries

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	101 VTargetSeries	None	NA	NA

Notes	Conversion Formula
- Series of target metadata. - Use the Target Local Set Tags. - The length field for VTargetSeries is the sum of the lengths of all the contained target metadata. - The length field for each VTarget Pack is the size of all elements within that pack including the TargetID.The length field for each VTarget Pack is the size of all elements within that set. - Up to 65535 instances of VTargetPack can reside within a VMTI LDS. The mandatory Target ID # allows discrimination between instances.	

Example Value	Example LDS Packet

UDS Key UDS Name	

Units	Range	Format
None	None	None

Notes

UDS Conversion

The VTargetSeries contains metadata pertaining to individual targets. One VTarget Pack is required for each reported target. Multiple instances (up to 65535) of the VTarget Pack can reside in one VMTI LDS pack. The mandatory Target ID # (first element of each VTarget Pack) provides discrimination between VTarget Packs.

9.13.0 Target ID Number

LDS Tag LDS Name UDS Mapped Key	None Target ID Number	Units	Range	Format
		None	1..65535	BER-OID encoded Value up to 65535

Notes	Conversion Formula
- This element is mandatory and it must come first in the VTarget Pack. It is BER-OID encoded to convey the length but has no Tag or Length field. - Range 1..65535 - A value of zero indicates an error.	x

Example Value	Example LDS Packet
`27`	`[V] = [0x1B]`

UDS Key UDS Name	

Units	Range	Format
None	1..65535	Uint16

Notes
The UDS Target set will include a 2 byte Target ID# element

UDS Conversion
UDS Target ID # element is BER-OID decoded LDS Target ID #

The target identification number uniquely identifies a target across multiple frames until the identification number is reset by the New Detection Flag (Tag 6 within the VTarget Pack). Sophisticated VMTI systems may use the same target ID number to identify a common target detected by different sensors, and therefore allow correlation between targets detected simultaneously by different sensor systems.

This element is mandatory as it has no Tag or Length field, thus its absence will result in incorrect parsing of the subsequent data. It shall be included as the first element of the VTargetLDS in the form of a BER-OID encoded single value.

9.13.1 Tag 1 Target Centroid Pixel Number

		Units	Range	Format
LDS Tag	1	Pixels	$1..2^{40}$-1	Unsigned Integer of Variable Length
LDS Name	Target Centroid Pixel Number			
UDS Mapped Key				

Notes	Conversion Formula
- Defines the position of the target within the video frame in pixels. - Range 1 to 2^{40}-1. - Numbering commences from one denoting the Top Left pixel (1,1) - Maximum size of frame is 1 Terapixel.	x

Example Value	Example LDS Packet
`27`	`[K] [L] [V] = [0d1][0d1][0x1B]`

UDS Key	
UDS Name	

Units	Range	Format
Pixels	$1..2^{40}$-1	Uint40

Notes

UDS Conversion

The X, Y position of the target within the frame is defined as a pixel number. The calculation of the pixel number uses the equation: X + ((Y-1) x Frame Width)). The top left pixel of the frame equates to a pixel number of 1. The frame width would normally be provided by the underlying video but in the absence of video or if the VMTI process is run on a different video input to that in the MPEG-2 TS then VMTI LDS Tag 7 can be used.

The range has been set at 5 bytes to allow frame sizes of up to 1 Terapixel. It is important for bit efficiency to rely on variable length payloads for this value. For example, if the pixel number is < 65536 then only two bytes should be used.

9.13.2 Tag 2 Target Bounding Box Top Left Pixel Number

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	2 Target Bounding Box Top Left Pixel Number	Pixels	1..2^{40}-1	Unsigned Integer of Variable Length

Notes	Conversion Formula
- Defines the position of the top left corner of the target bounding box within the video frame in pixels. - Range 1 to 2^{40}-1. - Numbering commences from one denoting the Top Left pixel - Maximum size of frame is 1 Terapixel.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d2][0d1][0x1B]

UDS Key UDS Name	

Units	Range	Format
Pixels	1..2^{40}-1	Uint40

Notes

UDS Conversion

The X, Y position of the top left corner of the bounding box is defined as a pixel number. The calculation of the pixel number uses the equation: X + ((Y-1) x Frame Width)). The top left pixel of the frame equates to a pixel number of 1. The frame width would normally be provided by the underlying video but in the absence of video or if the VMTI process is run on a different video input to that in the MPEG-2 TS then VMTI LDS Tag 7 can be used.

The range has been set at 5 bytes to allow frame sizes of up to 1 Terapixel. It is important for bit efficiency to rely on variable length payloads for this value. For example, if the pixel number is < 65536 then only two bytes should be used.

9.13.3 Tag 3 Target Bounding Box Bottom Right Pixel Number

LDS Tag	3	Units	Range	Format
LDS Name	Target Bounding Box Bottom Right Pixel Number	Pixels	1..2^{40}-1	Unsigned Integer of Variable Length
UDS Mapped Key				

Notes	Conversion Formula
- Defines the position of the bottom right corner of the target bounding box within the video frame in pixels. - Range 1 to 2^{40}-1. - Numbering commences from one denoting the Top Left pixel - Maximum size of frame is 1 Terapixel.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d3][0d1][0x1B]

UDS Key	
UDS Name	

Units	Range	Format
Pixels	1..2^{40}-1	Uint40

Notes

UDS Conversion

The X, Y position of the bottom right corner of the bounding box is defined as a pixel number. The calculation of the pixel number uses the equation: X + ((Y-1) x Frame Width)). The top left pixel of the frame equates to a pixel number of 1. The frame width would normally be provided by the underlying video but in the absence of video or if the VMTI process is run on a different video input to that in the MPEG-2 TS the VMTI LDS Tag 7 can be used.

The range has been set at 5 bytes to allow frame sizes of up to 1 Terapixel. It is important for bit efficiency to rely on variable length payloads for this value. For example, if the pixel number is < 65536 then only two bytes should be used.

9.13.4 Tag 4 Target Priority

LDS Tag LDS Name UDS Mapped Key	4 Target Priority	Units	Range	Format
		None	1..255	Uint8

Notes	Conversion Formula
- Priority or validity of target based on criteria within the VMTI system. The target(s) with the highest priority may not have the highest confidence value. - Potential for use in limited bandwidth scenarios to only send highest priority targets. - Range 1 to 255 where 1 is the highest priority. - Multiple targets may have the same priority. - Range is 1 to 255, where 1 is the highest priority.	x

Example Value	Example LDS Packet
`27`	`[K] [L] [V] = [0d4][0d1][0x1B]`

UDS Key UDS Name	

Units	Range	Format
None	1..255	Uint8

Notes

UDS Conversion

Target priority is designed to provide systems that are downstream from VMTI processors a means for intelligently culling targets for a given frame. For example, VMTI processors may generate thousands of hits. These may be used meaningfully in trackers (where bandwidth may not be an issue) however other clients may have more restrictive bandwidth limitations or not wish to overload systems with thousands of hits. For example, from an HMI perspective, it may be undesirable to clutter a situational awareness display with thousands of VMTI targets.

9.13.5 Tag 5 Target Confidence Value

		Units	Range	Format
LDS Tag	5	None	1..255	Uint8
LDS Name	Target Confidence Value			
UDS Mapped Key				

Notes	Conversion Formula
- Confidence value of target based on criteria within the VMTI system. The target(s) with the highest confidence may not have the highest priority value. - Potential for use in limited bandwidth scenarios to only send highest confidence targets. - Range 1 to 255 where 1 is the highest confidence. Multiple targets may have the same confidence value.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d5][0d1][0x1B]

UDS Key	
UDS Name	

Units	Range	Format
None	0..255	Uint8

Notes

UDS Conversion

Target priority and target confidence levels are designed to provide systems that are downstream from VMTI processors a means for intelligently assessing and culling targets for a given frame. A target may be detected with a high confidence; however be a low priority target.

9.13.6 Tag 6 New Detection Flag/ Target History

		Units	Range	Format
LDS Tag	6	Frames	0..65535	Unsigned Integer of Variable Length
LDS Name	New Detection Flag/ Target History			
UDS Mapped Key				

Notes	Conversion Formula
- The number of previous times the same target has been detected. - Range 0 to 65535 frames. Where a value of 0 denotes the target as a new detection. - Detections are not required to be in consecutive frames.	x

Example Value	Example LDS Packet
`27`	`[K] [L] [V] = [0d6][0d1][0x1B]`

UDS Key	
UDS Name	

Units	Range	Format
None	0..65535	Uint16

Notes

UDS Conversion

Primarily designed to indicate a new target/ reuse of a previous target tag number (VMTI LDS Tag 51 to 65585), but also provides the ability to indicate target persistence and may provide useful context when a target re-appears after not being detected for a significant time.

9.13.7 Tag 7 Percentage of Target Pixels

LDS Tag LDS Name UDS Mapped Key	7 Percentage of Target Pixels	Units	Range	Format
		None	1..100	Uint8

Notes	Conversion Formula
- The percentage of pixels within the bounding box that are detected to be target pixels rather than background pixels. - Range 1 to 100 where 100 signifies that the target completely fills the bounding box. - Values above 100 are invalid.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d7][0d1][0x1B]

UDS Key UDS Name	

Units	Range	Format
None	1..100	Uint8

Notes

UDS Conversion

This element can be used to provide information about how many of the pixels within the bounding box are target pixels. The use of the VMask, VObject or VFeature local data sets is recommended for cases where more detail about the target is required.

9.13.8 Tag 8 Target Color

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	8 Target Color	None	0..2^{24}-1	Uint24

Notes	Conversion Formula
- Dominant color of the target. For use when metadata is transmitted in the absence of the underlying video. - RGB color value. - VFeature LDS can be used for more comprehensive color information.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d8][0d1][0x1B]

UDS Key UDS Name	

Units	Range	Format
None	0..2^{24}-1	Uint24

Notes

UDS Conversion

This tag may be used for general mapping of any multispectral dataset to an RGB value, however it is primary designed to provide color information about a target without inclusion of the video. For example, the target is a red car. Mapping sensor systems from frequencies outside of the visible spectrum to an RGB color value is supported, however this may be a source of confusion. Even assigning a monochrome (8-bit) infrared value to RGB as a grayscale may be misleading to an end user who will interpret the signal as a color rather than an indication of relative intensity within a scene. Monochrome intensity values can be sent (up to 24 bits) using Tag 9 – Target Intensity.

9.13.9 Tag 9 Target Intensity

LDS Tag	9
LDS Name	Target Intensity
UDS Mapped Key	

Units	Range	Format
None	$0..2^{24}-1$	Unsigned Integer of Variable Length

Notes	Conversion Formula
- Dominant intensity of the target. For use when metadata is transmitted in the absence of the underlying video - Range $0..2^{24}-1$. - VFeature LDS can be used for more comprehensive temperature information.	x

Example Value	Example LDS Packet
27	[K] [L] [V] = [0d9][0d1][0x1B]

UDS Key	
UDS Name	

Units	Range	Format
None	$0..2^{24}-1$	Uint24

Notes

UDS Conversion

This element provides the scope to include the indicative or dominant intensity of the target at a dynamic range of up to 24 bits. Any sensor with a broad dynamic range may use this tag; however it is primarily designed for use with Infrared systems that may detect targets at higher than 8-bits per pixel dynamic range and transmit video at lower dynamic ranges (or not include video in the transport stream). Other details providing context or relating to accuracy and precision would be contained in the VFeature LDS.

9.13.10 Tag 10 Target Location Latitude Offset

LDS Tag LDS Name UDS Mapped Key	10 Target Location Latitude Offset	Units	Range	Format
		Degrees	+/- 19.2	Unsigned Integer of Variable Length

Notes

- Latitude offset for target from frame centre latitude (Standard 0601). Based on WGS84 ellipsoid.
- Use with Frame Center Latitude.
- Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees.
- Use -(2^{23}) as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25 meters at equator.
- Range: +/- 2136 km.

Conversion Formula

$$\text{VTarget\_LDS\_dec} = \left(\frac{\text{VTarget\_LDS\_range}}{\text{int\_range}} * \text{VTarget\_LDS\_int}\right) + \text{UAS\_ LDS\_23\_dec}$$

$$\text{VTarget\_LDS\_10\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_ LDS\_10}\right) + \text{UAS\_ LDS\_23\_dec}$$

Example Value	Example LDS Packet
	`[K] [L] [V] = [0d10][0d2][0xnn nn]`

UDS Key UDS Name	

Units	Range	Format
Degrees	+/- 90	Double

Notes

UDS Conversion

$$\text{UDS\_dec} = \left(\frac{19.2}{16777214} * \text{VTarget\_LDS\_int}\right) + \text{UAS\_ LDS\_23\_dec}$$

To UDS:
- UDS = (double)((19.2/0xFFFFFE * VTarget_LDS) + LDS_23_dec)

To LDS:
- LDS = (int24)round(0xFFFFFE/19.2 * (UDS - Frame_Center_LAT))

The target location has a real earth coordinate represented by a latitude-longitude pair. Target locations that lie above the horizon do not correspond to a point on the earth. Also, target locations may lie outside of the mapped range. Both cases should either not be reported, or be reported as an “error”.

The Target Latitude Offset is added to the Frame Centre Latitude metadata item from the parent 0601 packet to determine the Latitude of the target. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated motion imagery target location.

9.13.11 Tag 11 Target Location Longitude Offset

LDS Tag	11	Units	Range	Format
LDS Name	Target Location Longitude Offset	Degrees	+/- 19.2	Unsigned Integer of Variable Length
UDS Mapped Key				

Notes

- Longitude offset for target from frame centre longitude (Standard 0601). Based on WGS84 ellipsoid.
- Use with Frame Center Longitude.
- Map $-(2^{23}-1)..(2^{23}-1)$ to +/- 19.2 degrees.
- Use $-(2^{23})$ as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25 meters at equator.
- Range: +/- 2136 km.

Conversion Formula

$$\text{VTarget\_LDS\_dec} = \left(\frac{\text{VTarget\_LDS\_range}}{\text{int\_range}} * \text{VTarget\_LDS\_int}\right) + \text{UAS\_ LDS\_24\_dec}$$

$$\text{VTarget\_LDS\_11\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_ LDS\_11}\right) + \text{UAS\_ LDS\_24\_dec}$$

Example Value	Example LDS Packet
	`[K] [L] [V] = [0d11][0d2][0xnn nn]`

UDS Key UDS Name	

Units	Range	Format
Degrees	+/- 180	Double

Notes

UDS Conversion

$$\text{UDS\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_LDS\_int}\right) + \text{UAS\_ LDS\_24\_dec}$$

To UDS:
- UDS = (double)((19.2/0xFFFFFE * VTarget_LDS) + LDS_24_dec)

To LDS:
- LDS = (int24)round(0xFFFFFE/19.2 * (UDS - Frame_Center_LON))

The target location has a real earth coordinate represented by a latitude-longitude pair. Target locations that lie above the horizon do not correspond to a point on the earth. Also, target locations may lie outside of the mapped range. Both cases should either not be reported, or be reported as an “error”.

The Target Longitude Offset is added to the Frame Centre Longitude metadata item from the parent 0601 packet to determine the Longitude of the target. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated motion imagery target location.

9.13.12 Tag 12 Target Elevation

		Units	Range	Format
LDS Tag LDS Name UDS Mapped Key	12 Target Elevation	Meters	-900..19000	Unsigned Integer of Variable Length

Notes

- Terrain elevation at target location.
- Map 0..(216-1) to -900..19000 meters.
- Resolution: ~0.3 meters.

Conversion Formula

$$\text{VTarget\_LDS\_dec} = \left(\frac{\text{VTarget\_LDS\_range}}{\text{uint\_range}} * \text{VTarget\_LDS\_uint}\right) - \text{Offset}$$

$$\text{VTarget\_LDS\_12\_dec} = \left(\frac{19900}{65535} * \text{VTarget\_LDS\_12}\right) - 900$$

Example Value	Example LDS Packet
1275.381 Meters	[K] [L] [V] = [0d12][0d2][0x1B FC]

UDS Key UDS Name	

Units	Range	Format
Meters	-900..19000	Uint16

Notes

UDS Conversion

As its name suggests this is the elevation of the target (in meters).

9.13.13 Tag 13 Bounding Box Top Left Latitude Offset

LDS Tag LDS Name UDS Mapped Key	13 Bounding Box Top Left Latitude Offset	Units	Range	Format
		Degrees	+/- 19.2	Unsigned Integer of Variable Length

Notes

- Latitude offset for target from frame centre latitude (Standard 0601). Based on WGS84 ellipsoid.
- Use with Frame Center Latitude.
- Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees.
- Use -(2^{23}) as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25 meters at equator.
- Range: +/- 2136 km.

Conversion Formula

$$\text{VTarget\_LDS\_dec} = \left(\frac{\text{VTarget\_LDS\_range}}{\text{int\_range}} * \text{VTarget\_LDS\_int} \right) + \text{UAS\_LDS\_23\_dec}$$

$$\text{VTarget\_LDS\_13\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_LDS\_13} \right) + \text{UAS\_LDS\_23\_dec}$$

Example Value	Example LDS Packet
	`[K] [L] [V] = [0d13][0d2][0xnn nn]`

UDS Key UDS Name	

Units	Range	Format
Degrees	+/- 90	Double

Notes

UDS Conversion

$$\text{UDS\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_LDS\_int} \right) + \text{UAS\_LDS\_23\_dec}$$

To UDS:
- UDS = (double)((19.2/0xFFFFFE * VTarget_LDS) + LDS_23_dec)

To LDS:
- LDS = (int24)round(0xFFFFFE/19.2 * (UDS - Frame_Center_LAT))

The bounding box corners have a real earth coordinate represented by a latitude-longitude pair. Bounding box corners that lie above the horizon do not correspond to a point on the earth. Also, bounding box corners may lie outside of the mapped range. Both cases should either not be reported, or be reported as an “error”.

The Bounding Box Top Left Latitude Offset is added to the Frame Centre Latitude metadata item from the parent 0601 packet to determine the Latitude of the top left corner of the target bounding box. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated motion imagery bounding box corner location.

9.13.14 Tag 14 Bounding Box Top Left Longitude Offset

		Units	Range	Format
LDS Tag	14	Degrees	+/- 19.2	Unsigned Integer of Variable Length
LDS Name	Bounding Box Top Left Longitude Offset			
UDS Mapped Key				

Notes

- Longitude offset for target from frame centre longitude (Standard 0601). Based on WGS84 ellipsoid.
- Use with Frame Center Longitude.
- Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees.
- Use -(2^{23}) as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25 meters at equator.
- Range: +/- 2136 km.

Conversion Formula

$$\mathrm{VTarget\_LDS\_dec} = \left(\frac{\mathrm{VTarget\_LDS\_range}}{\mathrm{int\_range}} * \mathrm{VTarget\_LDS\_int}\right) + \mathrm{UAS\_\ LDS\_24\_dec}$$

$$\mathrm{VTarget\_LDS\_14\_dec} = \left(\frac{19.2}{16777216} * \mathrm{VTarget\_\ LDS\_14}\right) + \mathrm{UAS\_\ LDS\_24\_dec}$$

Example Value	Example LDS Packet
	`[K] [L] [V] = [0d14][0d2][0xnn nn]`

UDS Key UDS Name	

Units	Range	Format
Degrees	+/- 180	Double

Notes

UDS Conversion

$$\mathrm{UDS\_dec} = \left(\frac{19.2}{16777216} * \mathrm{VTarget\_LDS\_int}\right) + \mathrm{UAS\_\ LDS\_24\_dec}$$

To UDS:
- UDS = (double)((19.2/0xFFFFFE * VTarget_LDS) + LDS_24_dec)

To LDS:
- LDS = (int24)round(0xFFFFFE/19.2 * (UDS - Frame_Center_LON))

The bounding box corners have a real earth coordinate represented by a latitude-longitude pair. Bounding box corners that lie above the horizon do not correspond to a point on the earth. Also, bounding box corners may lie outside of the mapped range. Both cases should either not be reported, or be reported as an “error”.

The Bounding Box Top Left Longitude Offset is added to the Frame Centre Longitude metadata item from the parent 0601 packet to determine the Longitude of the top left corner of the target bounding box. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated motion imagery bounding box corner location.

9.13.15 Tag 15 Bounding Box Bottom Right Latitude Offset

		Units	Range	Format
LDS Tag	15			
LDS Name	Bounding Box Bottom Right Latitude Offset	Degrees	+/- 19.2	Unsigned Integer of Variable Length
UDS Mapped Key				

Notes

- Latitude offset for target from frame centre latitude (Standard 0601). Based on WGS84 ellipsoid.
- Use with Frame Center Latitude.
- Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees.
- Use -(2^{23}) as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25 meters at equator.
- Range: +/- 2136 km.

Conversion Formula

$$\text{VTarget\_LDS\_dec} = \left(\frac{\text{VTarget\_LDS\_range}}{\text{int\_range}} * \text{VTarget\_LDS\_int}\right) + \text{UAS\_ LDS\_23\_dec}$$

$$\text{VTarget\_LDS\_15\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_ LDS\_15}\right) + \text{UAS\_ LDS\_23\_dec}$$

Example Value	Example LDS Packet
	`[K] [L] [V] = [0d15][0d2][0xnn nn]`

UDS Key UDS Name	

Units	Range	Format
Degrees	+/- 90	Double

Notes

UDS Conversion

$$\text{UDS\_dec} = \left(\frac{19.2}{16777216} * \text{VTarget\_LDS\_int}\right) + \text{UAS\_ LDS\_23\_dec}$$

To UDS:
- UDS = (double)((19.2/0xFFFFFE * VTarget_LDS) + LDS_23_dec)

To LDS:
- LDS = (int24)round(0xFFFFFE/19.2 * (UDS - Frame_Center_LAT))

The bounding box corners have a real earth coordinate represented by a latitude-longitude pair. Bounding box corners that lie above the horizon do not correspond to a point on the earth. Also, bounding box corners may lie outside of the mapped range. Both cases should either not be reported, or be reported as an “error”.

The Bounding Box Bottom Right Latitude Offset is added to the Frame Centre Latitude metadata item from the parent 0601 packet to determine the Latitude of the bottom right corner of the target bounding box. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated motion imagery bounding box corner location.

9.13.16 Tag 16 Bounding Box Bottom Right Longitude Offset

		Units	Range	Format
LDS Tag	16	Degrees	+/- 19.2	Unsigned Integer of Variable Length
LDS Name	Bounding Box Bottom Right Longitude Offset			
UDS Mapped Key				

Notes

- Longitude offset for target from frame centre longitude (Standard 0601). Based on WGS84 ellipsoid.
- Use with Frame Center Longitude.
- Map -(2^{23}-1)..(2^{23}-1) to +/- 19.2 degrees.
- Use -(2^{23}) as an "error" indicator..
- Resolution: ~1.2micro deg, ~0.25 meters at equator.
- Range: +/- 2136 km.

Conversion Formula

$$\mathrm{VTarget\_LDS\_dec} = \left(\frac{\mathrm{VTarget\_LDS\_range}}{\mathrm{int\_range}} * \mathrm{VTarget\_LDS\_int}\right) + \mathrm{UAS\_LDS\_24\_dec}$$

$$\mathrm{VTarget\_LDS\_16\_dec} = \left(\frac{19.2}{16777216} * \mathrm{VTarget\_LDS\_16}\right) + \mathrm{UAS\_LDS\_24\_dec}$$

Example Value	Example LDS Packet
	`[K] [L] [V] = [0d16][0d2][0xnn nn]`

UDS Key UDS Name	

Units	Range	Format
Degrees	+/- 180	Double

Notes

UDS Conversion

$$\mathrm{UDS\_dec} = \left(\frac{19.2}{16777216} * \mathrm{VTarget\_LDS\_int}\right) + \mathrm{UAS\_LDS\_24\_dec}$$

To UDS:
- UDS = (double)((19.2/0xFFFFFE * VTarget_LDS) + LDS_24_dec)

To LDS:
- LDS = (int24)round(0xFFFFFE/19.2 * (UDS - Frame_Center_LON))

The bounding box corners have a real earth coordinate represented by a latitude-longitude pair. Bounding box corners that lie above the horizon do not correspond to a point on the earth. Also, bounding box corners may lie outside of the mapped range. Both cases should either not be reported, or be reported as an “error”.

The Bounding Box Bottom Right Longitude Offset is added to the Frame Centre Longitude metadata item from the parent 0601 packet to determine the Longitude of the bottom right corner of the target bounding box. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated motion imagery bounding box corner location.

9.13.17 Tag 101 VMask LDS

9.13.17.1

Place first VMask LDS element here – subsequent elements 9.13.17.x

9.13.18 Tag 102 VObject LDS

9.13.18.1

Place first VObject LDS element here – subsequent elements 9.13.18.x

9.13.19 Tag 103 VFeature LDS

9.13.19.1

Place first VFeature LDS element here – subsequent elements 9.13.19.x

9.13.20 Tag 104 VTracker LDS

9.13.20.1

Place first VTracker LDS element here – subsequent elements 9.13.20.x